

A General Metric for Riemannian Manifold Hamiltonian Monte Carlo

Michael Betancourt

*Department of Statistics, Columbia University, New York, NY 10027, USA**

(Dated: December 20, 2012)

Markov Chain Monte Carlo (MCMC) is an invaluable means of inference with complicated models, and Hamiltonian Monte Carlo, in particular Riemannian Manifold Hamiltonian Monte Carlo (RMHMC), has demonstrated impressive success in many challenging problems. Current RMHMC implementations, however, rely on a Riemannian metric that limits their application to analytically-convenient models. In this paper I propose a new metric for RMHMC without these limitations and verify its success on a distribution that emulates many hierarchical and latent models.

Riemannian Manifold Hamiltonian Monte Carlo provides a powerful tool for the efficient sampling from complex distributions, but the applicability of existing approaches has been limited by the dependency on the Fisher-Rao metric. In this paper I introduce a new metric that admits a general implementation of Riemannian Manifold Hamiltonian Monte Carlo and demonstrate its efficacy on a distribution that mirrors the pathological behavior of common models.

I. HAMILTONIAN MONTE CARLO

Hamiltonian Monte Carlo (HMC) takes advantage of symplectic geometry to yield efficient Markov transitions [1]. Augmenting the parameters of an N -dimensional target density, $\pi(\mathbf{q})$, with corresponding momenta, \mathbf{p} , defines a joint density,

$$\begin{aligned}\pi(\mathbf{p}, \mathbf{q}) &= \pi(\mathbf{p}|\mathbf{q}) \pi(\mathbf{q}) \\ &= \exp[\log \pi(\mathbf{p}|\mathbf{q})] \exp[\log \pi(\mathbf{q})] \\ &\propto \exp[-T(\mathbf{p}, \mathbf{q})] \exp[-V(\mathbf{q})] \\ &\equiv \exp[-H(\mathbf{p}, \mathbf{q})].\end{aligned}$$

The Hamiltonian, $H(\mathbf{p}, \mathbf{q}) = T(\mathbf{p}, \mathbf{q}) + V(\mathbf{q})$, defines trajectories between points $\mathbf{z} = \{\mathbf{p}, \mathbf{q}\}$ via the differential equations

$$\begin{aligned}\frac{d\mathbf{q}}{dt} &= +\frac{\partial H}{\partial \mathbf{p}} \\ \frac{d\mathbf{p}}{dt} &= -\frac{\partial H}{\partial \mathbf{q}}.\end{aligned}$$

Because these trajectories preserve the value of the Hamiltonian and the differential volume $d^{2N}\mathbf{z}$, they also define Markovian transitions with the stationary density $\pi(\mathbf{p}, \mathbf{q})$. Alternating this Hamiltonian evolution with conditional samples of the momenta,

$$\mathbf{p} \sim \pi(\mathbf{p}|\mathbf{q}) \propto \exp[-T(\mathbf{p}, \mathbf{q})],$$

yields an ergodic Markov chain sampling from \mathbf{z} and, because the marginal of $\pi(\mathbf{p}, \mathbf{q})$ is constructed to be the

target distribution, the desired samples from $\pi(\mathbf{q})$ follow by simply disregarding the momenta.

No matter the choice of the kinetic energy, $T(\mathbf{p}, \mathbf{q})$, the evolution equations incorporate the gradient of the potential, $V(\mathbf{q})$, and hence higher order information about the target distribution. This gradient guides the Markov chain along regions of high probability mass and reduces random walk behavior. Note that, in practice, the Hamiltonian evolution cannot be performed analytically and we must resort to numerical integration. Error in the integration scheme introduces bias into the transitions, but this is readily avoided by considering the evolution not as a transition but rather as the proposal for a Metropolis transition [2, 3].

The first [2] and still most common choice of the conditional density, $\pi(\mathbf{p}|\mathbf{q})$, is a standard gaussian,

$$\pi(\mathbf{p}|\mathbf{q}) = \mathcal{N}(\mathbf{p}|\mathbf{0}, \mathbf{M}),$$

or

$$T(\mathbf{p}, \mathbf{q}) = \frac{1}{2} \mathbf{p}^T \cdot \mathbf{M}^{-1} \cdot \mathbf{p}, \quad (1)$$

where the mass matrix \mathbf{M} allows for a global decorrelation and rescaling of the parameters with respect to each other. This choice, however, ultimately limits the effectiveness of HMC when applied to intricate target distributions. Because $\mathbf{p}^T \cdot \mathbf{M}^{-1} \cdot \mathbf{p}$ is a χ^2 variate, in equilibrium $\Delta T \approx N/2$ and, with the Hamiltonian conserved along each trajectory, this implies that the variation in the potential is also limited to $\Delta V \approx N/2$. When the target distribution is highly correlated, the typical set spans a potential gap much larger than this: the resulting samples become highly correlated no matter how long the trajectories are evolved [3] and the Markov chain devolves towards a random walk.

Another issue with the simple choice (1) is that the inevitable numerical integration introduces a spatial scale into the system via a finite step-size. Complicated target distributions will typically exhibit multiple spatial scales depending on the particular value of the parameters, and any single choice of a step-size will generate at least some inefficiency. If the step-size is chosen to maximize efficiency, as common in adaptive schemes, regions of the target distribution with large curvature, and hence small spatial scales, can be missed entirely by the numerical trajectories.

* betanalpha@gmail.com

These weaknesses can be overcome by appealing to a more sophisticated choice of the conditional density: a gaussian conditionally dependent on the \mathbf{q} through a covariance matrix,

$$\pi(\mathbf{p}|\mathbf{q}) = \mathcal{N}(\mathbf{p}|\mathbf{0}, \Sigma(\mathbf{q})),$$

or

$$T(\mathbf{p}, \mathbf{q}) = \frac{1}{2} \mathbf{p}^T \cdot \Sigma^{-1}(\mathbf{q}) \cdot \mathbf{p} + \frac{1}{2} \log |\Sigma(\mathbf{q})|.$$

Because the resulting Hamiltonian trajectories are related to geodesics on a Riemannian manifold with metric $\Sigma(\mathbf{q})$, this choice is known as *Riemannian Manifold Hamiltonian Monte Carlo* (RMHMC) [4]. Similarly, the constant metric of (1) can be thought of as emulating dynamics on a Euclidean manifold, and to be consistent I will refer to use of the simpler Hamiltonian as *Euclidean Manifold Hamiltonian Monte Carlo* (EMHMC).

The freedom in specifying a metric admits two significant improvements: a proper choice of $\Sigma(\mathbf{q})$ can dynamically decorrelate and rescale the target distribution to avoid inefficiencies in the numerical integration, while also yielding a dynamic determinant whose variations can compensate for much larger variations in the potential.

What, however, exactly defines a proper choice for the metric? When the target distribution is a multivariate gaussian,

$$V(\mathbf{q}) = \frac{1}{2} \mathbf{q}^T \cdot \mathbf{S}^{-1} \cdot \mathbf{q},$$

the target distribution is standardized by taking $\Sigma(\mathbf{q}) = \mathbf{S}^{-1}$ [3]. In a convex neighborhood any target distribution can be approximated by a multivariate gaussian,

$$\pi(\mathbf{q}) \approx \mathcal{N}(\mathbf{q}|\mathbf{0}, \mathbf{H}^{-1})$$

or, equivalently,

$$V(\mathbf{q}) \approx \frac{1}{2} \mathbf{q}^T \cdot \mathbf{H} \cdot \mathbf{q},$$

with the Hessian matrix

$$H_{ij} = \partial^2 V / \partial q^i \partial q^j,$$

which immediately motivates the candidate metric $\Sigma(\mathbf{q}) = \mathbf{H}$.

This metric quickly runs into problems, however, when the target distribution is not globally convex. In neighborhoods where the Hessian is not positive-definite, for example, the conditional density $\pi(\mathbf{p}|\mathbf{q})$ becomes improper. Moreover, in the neighborhoods where the signature of the Hessian changes, the log determinant diverges and the Hamiltonian evolution becomes singular. These neighborhoods effectively partition the support of the target distribution into a disjoint union of compact neighborhoods between which the Markov chain cannot transition.

One way to avoid indefinite metrics is to take advantage of any conditioning variables, \mathbf{y} , in the target distribution. Marginalizing the Hessian over these conditioning variables yields the Fisher-Rao metric [5],

$$\Sigma_{ij} = \mathbb{E}_{\mathbf{y}} \left[\frac{\partial^2 V(\mathbf{q}|\mathbf{y})}{\partial q^i \partial q^j} \right],$$

which is guaranteed to be positive-semidefinite. For all but the simplest conditional distributions, however, the marginalization is unfeasible and, even when it can be performed analytically, the resulting metric can still be singular. Moreover, the marginalization removes the correlation between variables in many hierarchical and latent models, almost eliminating the effectiveness of the metric. Of course, all of this is immaterial if the target distribution lacks natural conditioning variables.

We need a means of constructing a metric from the Hessian that is not only everywhere well-behaved but also practical to compute for any given target distribution.

II. THE SOFTABS METRIC

With a careful application of matrix functions, it is possible to maintain the desirable behavior of the Hessian in convex neighborhoods while avoiding its singular behavior elsewhere. Moreover, because the functions are local the resulting metric is readily implemented for general distributions.

A. Definition

The exponential map [6], \exp , is a matrix function from the space of all matrices to the component of the general linear group, $\text{GL}(n)$, connected to the identity matrix: an isomorphism of the space of positive-definite matrices. Because this mapping preserves the symmetric part of the domain, any symmetric matrix, such as the Hessian, is guaranteed to be mapped to a symmetric, positive-definite matrix admissible as a Riemannian metric.

One benefit of the exponential map is that it preserves the eigenbasis of the input matrix, \mathbf{X} . If

$$\mathbf{X} = \mathbf{Q} \cdot \boldsymbol{\lambda} \cdot \mathbf{Q}^T$$

is the eigendecomposition of \mathbf{X} with $\boldsymbol{\lambda} = \text{Diag}(\lambda_i)$ the diagonal matrix of eigenvalues and \mathbf{Q} the corresponding matrix of eigenvectors, then the exponential map yields

$$\exp \mathbf{X} = \mathbf{Q} \cdot \exp \boldsymbol{\lambda} \cdot \mathbf{Q}^T.$$

The metric $\exp \mathbf{H}$ provides the same decorrelation as the Hessian but, unfortunately, also severely warps the eigenvalues and the corresponding rescaling of the local parameters.

By combining multiple exponential mappings, however, we can largely preserve the spectral decomposition

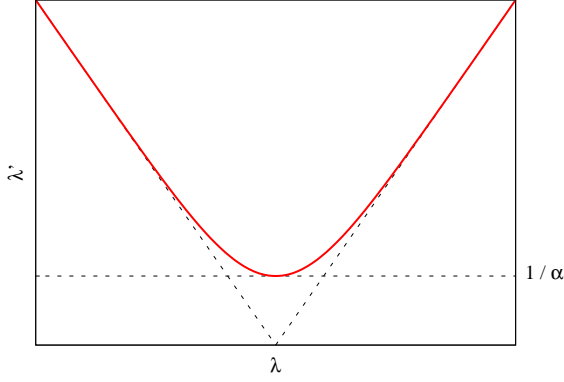


FIG. 1. The SoftAbs map preserves the eigenbasis of the Hessian but transforms the eigenvalues, λ , with a smooth approximation to the absolute value. The inverse of the regularization parameter, α , controls the “hardness” of the approximation; as $\alpha \rightarrow \infty$ the SoftAbs map reduces to the exact absolute value.

of the Hessian. In particular, the *SoftAbs* map

$$\mathfrak{X} \equiv [\exp(\alpha \mathbf{X}) + \exp(-\alpha \mathbf{X})] \cdot \mathbf{X} \cdot [\exp(\alpha \mathbf{X}) - \exp(-\alpha \mathbf{X})]^{-1}$$

approximates the absolute value of the eigenspectrum with a smooth function:

$$\mathfrak{X} = \mathbf{Q} \cdot \mathfrak{X} \cdot \mathbf{Q}^T,$$

where

$$\begin{aligned} \mathfrak{X} &= \text{Diag} \left(\lambda_i \frac{e^{\alpha \lambda_i} + e^{-\alpha \lambda_i}}{e^{\alpha \lambda_i} - e^{-\alpha \lambda_i}} \right) \\ &= \text{Diag} (\lambda_i \coth \alpha \lambda_i). \end{aligned}$$

This map not only ensures that the transformed eigenvalues are positive but also regularizes any small eigenvalues that might introduce numerical instabilities (Figure 1).

Applying the SoftAbs map to the Hessian guarantees a well-behaved metric for RMHMC, \mathfrak{H} , that preserves the desired properties of the Hessian while regularizing its numerical singularities. In a practical implementation, α limits the scaling of the integration step-size and restrains the numerical integrator from unwise extrapolations, emulating a trust region common in nonlinear optimization [7].

This construction also motivates a family of approximate metrics with possible utility in circumstances of limited computational resources (Appendix A).

B. Implementation

In practice, exponential maps can be difficult to implement [8]; the eigendecomposition used above, for example, can suffer from numerical instabilities when applied to general matrices because of ambiguities among the

eigenvectors. The Hessian, however, is symmetric and the eigenvectors are guaranteed to be orthogonal. Consequently, the eigendecomposition is well-behaved and provides a practical means of computing the SoftAbs map.

To implement the SoftAbs metric we first perform the eigendecomposition of the Hessian

$$\mathbf{H} = \mathbf{Q} \cdot \boldsymbol{\lambda} \cdot \mathbf{Q}^T,$$

and then reconstruct the metric as

$$\mathfrak{H} = \mathbf{Q} \cdot \mathfrak{X} \cdot \mathbf{Q}^T,$$

with $\mathfrak{X} = \text{Diag} (\lambda_i \coth \alpha \lambda_i)$.

Hamiltonian evolution also requires two derivatives: the gradient of the quadratic form, $\mathbf{p}^T \cdot \mathfrak{H}^{-1} \cdot \mathbf{p}$, and the log determinant, $\log |\mathfrak{H}|$.

The latter can be computed as [9, 10]

$$\begin{aligned} \partial \left(\mathbf{p}^T \cdot \mathfrak{H}^{-1} \cdot \mathbf{p} \right) &= \mathbf{p}^T \cdot \partial \mathfrak{H}^{-1} \cdot \mathbf{p} \\ &= -\mathbf{p}^T \cdot \mathfrak{H}^{-1} \cdot \partial \mathfrak{H} \cdot \mathfrak{H}^{-1} \cdot \mathbf{p} \\ &= -(\mathbf{Q}^T \cdot \mathbf{p})^T [\mathbf{J} \circ \mathbf{Q}^T \cdot \partial \mathbf{H} \cdot \mathbf{Q}] (\mathbf{Q}^T \cdot \mathbf{p}), \end{aligned}$$

where \circ denotes the Hadamard product and

$$J_{ij} \equiv \frac{\lambda_i \coth \alpha \lambda_i - \lambda_j \coth \alpha \lambda_j}{\lambda_i - \lambda_j}.$$

Note that when $\lambda_i = \lambda_j$, such as for the diagonal elements or degenerate eigenvalues, this becomes the derivative,

$$J_{ij} \rightarrow \frac{\partial}{\partial \lambda_i} \lambda_i \coth \alpha \lambda_i.$$

Unfortunately, this form of the gradient is computationally inefficient, requiring $O(N^3)$ for each component of the gradient, and hence $O(N^4)$ overall. Taking advantage of the properties of the Hadamard product [11], however, the gradient can be manipulated to give

$$\partial \left(\mathbf{p}^T \mathfrak{H}^{-1} \mathbf{p} \right) = -\text{Tr} [\mathbf{Q} \cdot \mathbf{D} \cdot \mathbf{J} \cdot \mathbf{D} \cdot \mathbf{Q}^T \cdot \partial \mathbf{H}],$$

where $\mathbf{D} = \text{Diag} ((\mathbf{Q}^T \cdot \mathbf{p})_i)$. If the matrix $\mathbf{Q} \cdot \mathbf{D} \cdot \mathbf{J} \cdot \mathbf{D} \cdot \mathbf{Q}^T$ is first cached, then each component of the gradient can be computed in only $O(N^2)$ so that the complete gradient does not exceed the $O(N^3)$ complexity of the decomposition itself.

Similar Hadamard identities reduce the gradient of the log determinant to

$$\partial \log |\mathfrak{H}| = \text{Tr} [\mathbf{Q} (\mathbf{R} \circ \mathbf{J}) \mathbf{Q}^T \cdot \partial \mathbf{H}],$$

where

$$\mathbf{R} = \text{Diag} \left(\frac{1}{\lambda_i \coth \alpha \lambda_i} \right).$$

Once again, caching the intermediate matrix, $\mathbf{Q} (\mathbf{R} \circ \mathbf{J}) \mathbf{Q}^T$, enables the full gradient to be computed in $O(N^3)$.

These results admit an efficient symplectic integrator (Appendix B) for RMHMC; a C++ implementation is available online at <http://betanalpha.github.com/jamon/>.

Algorithm	Warm-Up Iterations	Samples	ϵ	Accept Rate	CPU Time (s)	ESS	ESS / Time (s ⁻¹)
EMHMC	10 ³	10 ⁵	0.001	0.999	1627	70.3	0.0432
RMHMC	10 ³	10 ³	0.21	0.946	6282	856	0.136

TABLE I. When comparing the effective sample size of the latent variable, v , in the funnel distribution, hand-tuned EMHMC is over three times less effective than adaptively-tuned RMHMC. CPU time was measured with the `clock` function in the C++ library `time`.

III. EXPERIMENTS

The utility of the SoftAbs metric is best demonstrated on complex distributions. Neal’s funnel distribution [12]

$$\pi(\mathbf{x}, v) = \prod_{i=1}^n \mathcal{N}(x_i | 0, e^{-v}) \cdot \mathcal{N}(v | 0, 9),$$

emulates many pathological features of popular distributions, such as those arising in hierarchical [13] and latent [14] models. Note that, by construction, the marginal distribution of v is simply $v \sim \mathcal{N}(0, 9)$ ¹, independent of n , admitting v and its marginal distribution as a simple diagnostic of bias in any sampling procedure.

In each experiment a Markov chain is randomly initialized, $q_i \sim U(-1, 1)$, and then taken through a series of warm-up iterations before sampling begins. Where noted, the integrator step-size, ϵ , is adapted with dual-averaging to yield a target Metropolis acceptance rate [15]. The number of integration steps is set by hand to approximate the half-period of the oscillating trajectories (Figures 3, 5).

Autocorrelations, ρ_i , of v are computed with an initial monotone sequence estimator [16] and the effective sample size (ESS) is defined as $\text{ESS} = I \left(1 + 2 \sum_{i=1}^I \rho_i \right)^{-1}$, where I is the total number of generated samples.

The above procedure is applied to EMHMC with step-size adaptation, EMHMC without step-size adaption, and RMHMC with the SoftAbs metric.

A. EMHMC with Adaptation

Despite its simplicity, the funnel demonstrates many of the limitations of EMHMC. When adaptively tuned to the nominal acceptance rate $r = 0.65$ [3], the integrator step-size exceeds the spatial scale of the narrow neck; even though the probability mass of the mouth and neck of the funnel is comparable, the resulting trajectories overlook the neck entirely and bias resulting expectations without any obvious indication (Figure 2).

B. EMHMC without Adaptation

Because we know the truth in this case, we can abandon adaptive tuning and instead tune the step-size by hand; a smaller step-size ensures that the trajectories explore most of the funnel’s probability mass and that the marginal distribution $p(v)$ is correct within Monte Carlo error. Unfortunately, the funnel also exhibits the limitations of a position-independent kinetic energy. The variation of the potential within the typical set is huge, and the meager variation of the kinetic energy dramatically restricts the distance of each transition (Figure 3). The EMHMC transitions struggle to cross between the mouth and neck of the funnel, and the Markov chain becomes little more than a random walk across the distribution (Figure 4).

C. RMHMC with the SoftAbs Metric

On the other hand, the SoftAbs metric, here with $\alpha = 10^6$, allows RMHMC to explore the entire distribution within a single trajectory (Figure 5). Because the metric accounts for local curvature, the step-size can be adaptively tuned² without introducing any bias. The huge autocorrelations of EMHMC vanish (Figure 6) and, despite the increased computation required for each transition, RMHMC yields a more efficient generation of effective samples (Table I).

Nominally, the $O(N^3)$ computational burden of RMHMC is significantly worse than the $O(N)$ burden of EMHMC. The pathological behavior of distributions like the funnel, however, scales much faster, often exponentially, and the benefit of RMHMC with the SoftAbs metric only increases with dimension. Moreover, this concern ignores the burden of computing the potential itself which, as in the case of Bayesian posteriors with many data, can overwhelm the $O(N^3)$ burden entirely.

¹ Note the use of the convention $\mathcal{N}(\mu, \sigma^2)$.

² The increased information encoded in the metric should admit a larger acceptance rate, r , for RMHMC than the EMHMC case of $r = 0.65$. Motivated by some simple experiments, here the target rate for RMHMC is set to $r = 0.95$.

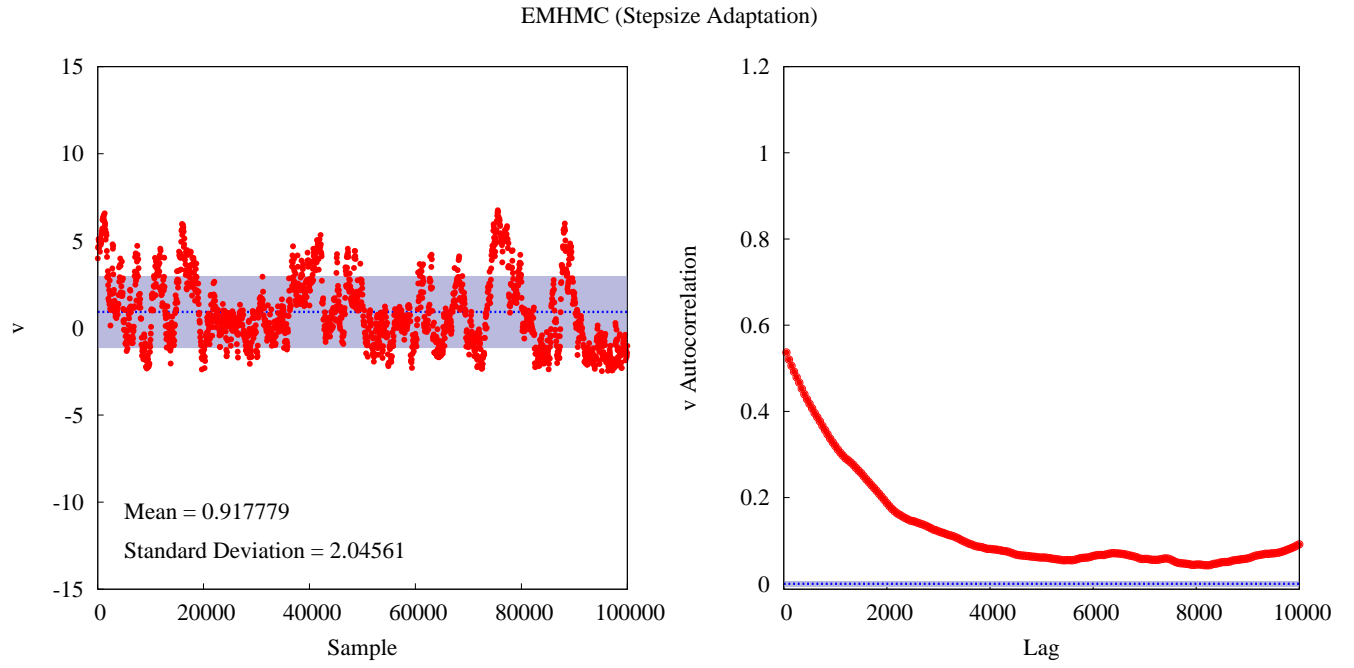


FIG. 2. Adaptive tuning of EMHMC results in an excessively large step-size, preventing trajectories from exploring the neck of the funnel and biasing the stationary distribution. Consequently, the samples of v are inconsistent with the marginal distribution $\mathcal{N}(0, 9)$.

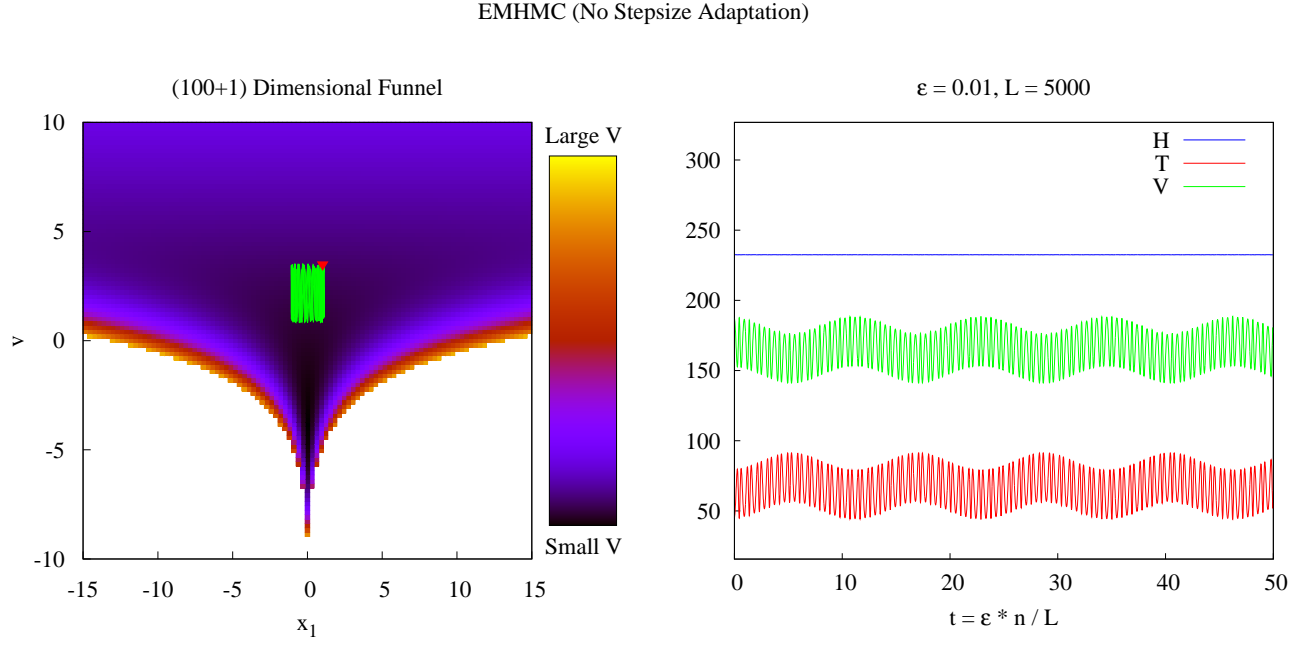


FIG. 3. EMHMC trajectories are limited to $\Delta V \sim (n+1)/2$ and consequently explore only a small neighborhood of the funnel. Note that the trajectories are oscillatory; half of the largest period of oscillation, here $T/2 \approx 8$, defines the optimal integration time for maximizing distance, and minimizing autocorrelation, between the samples.

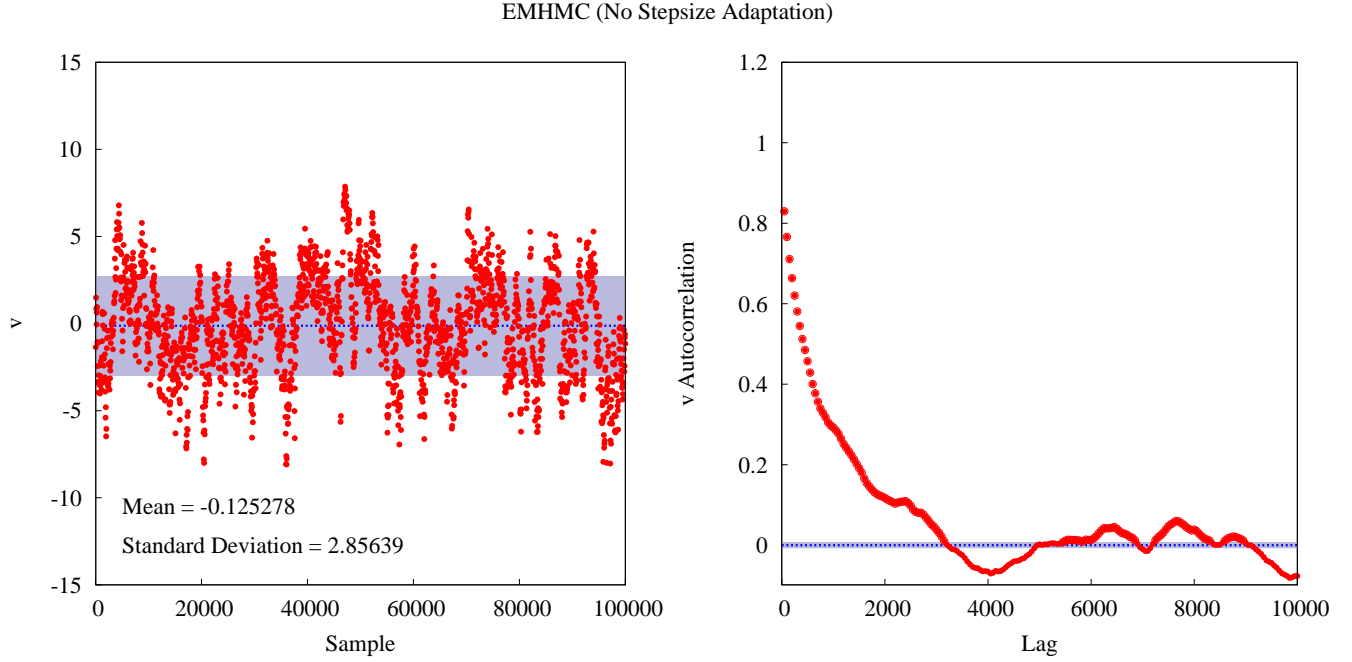


FIG. 4. Although not optimal, a smaller step-size tuned by hand allows the trajectories to penetrate the neck of the funnel, and the resulting samples of v are consistent with the true marginal, $\mathcal{N}(0, 9)$. In real applications where the truth is not known a priori, this sort of hand-tuning is not viable.

RMHMC with SoftAbs Metric

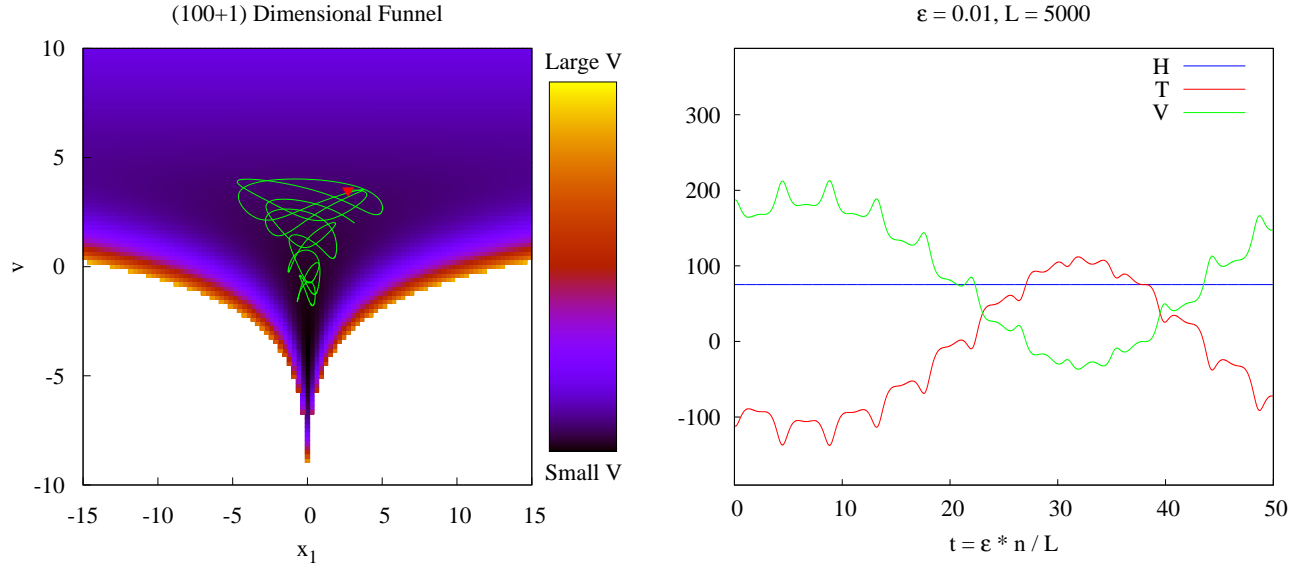


FIG. 5. The log determinant in the Hamiltonian allows RMHMC trajectories to explore without the restriction to $\Delta V \sim (n+1)/2$, and the dynamic decorrelation/scaling ensures that a single integrator step-size is efficient across the entire distribution. As in EMHMC, the trajectories oscillate and the half-period of the longest oscillation, $T/2 \approx 25$, defines the optimal integration time.

RMHMC with SoftAbs Metric

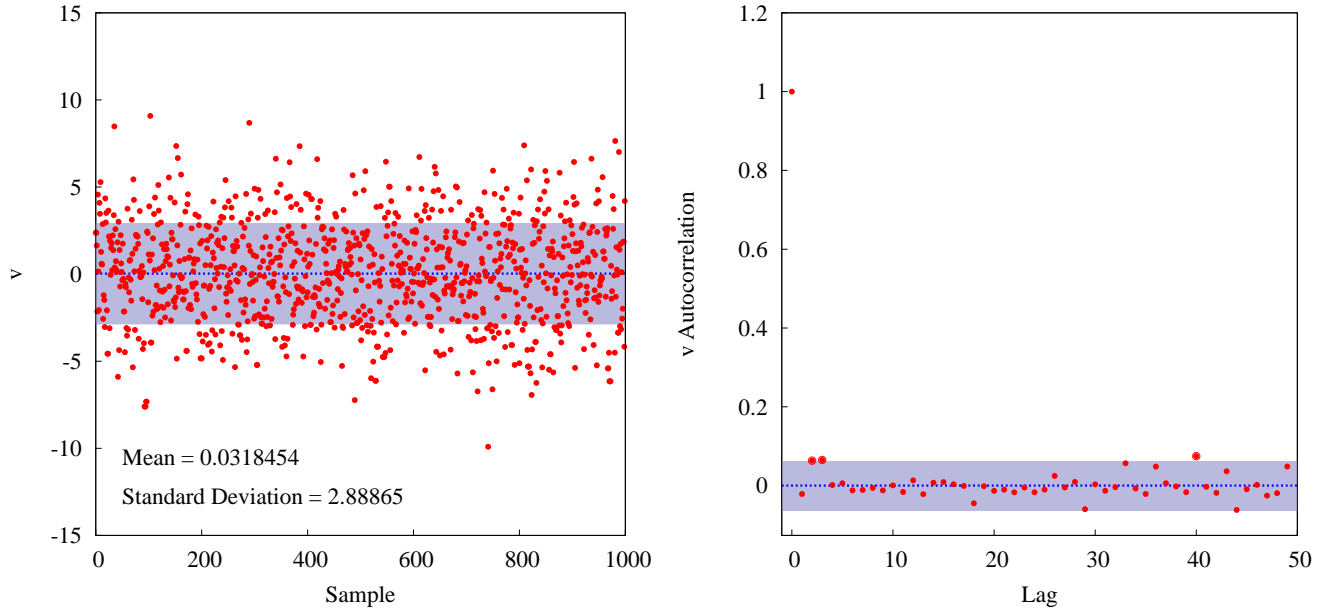


FIG. 6. The far-reaching trajectories of RMHMC dramatically reduce the autocorrelation of the samples which, despite the step-size adaptation, are consistent with the desired marginal $\mathcal{N}(0, 9)$. Note that significantly fewer lags shown here than above.

IV. CONCLUSIONS AND FUTURE WORK

By smoothly regularizing the eigendecomposition of the Hessian, the SoftAbs metric admits a general implementation of RMHMC robust against the many pathologies to which EMHMC can be vulnerable. Despite its apparently steep computational burden, the SoftAbs metric allows for practical inference on complex models never before deemed feasible.

Potential to reduce that burden might be found by looking deeper into Riemannian geometry. Use of frame bundles [17] to transport the Hessian eigenbasis along a trajectory, for example, could be more efficient than recomputing the eigendecomposition at each point.

Moreover, further insight into the metric itself may be found in understanding the geometric consequences of the SoftAbs mapping. What effect, for example, does the regularization of the SoftAbs mapping have on the geometric curvature of the manifold? Perhaps this parameterized regularization be related to Ricci flow [18], or other smoothing diffeomorphisms.

V. ACKNOWLEDGEMENTS

I warmly thank Bob Carpenter, Joe Formaggio, Mark Girolami, and Chris Jones for helpful comments and suggestions, as well as Andrew Gelman and the Stan team for their generous hospitality.

APPENDIX A: APPROXIMATIONS TO THE SOFTABS METRIC

A. Approximations

The SoftAbs metric requires an eigendecomposition and all third-order derivatives of the potential, both of which can be sufficiently burdensome as to render the metric unfeasible for very large problems. Transforming various approximations to the Hessian with the SoftAbs map, however, yields a series of approximations that offer less intense alternatives that may be useful for certain distributions.

1. Diagonal

Ignoring the local decorrelation of the potential and instead focusing on the rescaling of the parameters, we can simply take

$$\mathbf{H} \approx \text{Diag}(H_{ii}),$$

with

$$\mathfrak{H} \approx \text{Diag}(H_{ii} \coth \alpha H_{ii}).$$

The computational burden of the resulting evolution scales as $O(N^2)$, much better than the $O(N^3)$ of the full SoftAbs metric, and, because the Hessian of the funnel is almost diagonal, the approximation loses little information. Consequently, the diagonal approximation performs exceptionally well in this case (Table II, Figures 7, 8). Note that, on more correlated distributions the approximation will be more limiting and the performance will suffer.

Algorithm	Warm-Up Iterations	Samples	ϵ	Accept Rate	CPU Time (s)	ESS	ESS / Time (s ⁻¹)
EMHMC	10 ³	10 ⁵	0.001	0.999	1627	70.3	0.0432
RMHMC (SoftAbs)	10 ³	10 ³	0.21	0.946	6282	856	0.136
RMHMC (Diag SoftAbs)	10 ³	10 ³	0.49	0.805	7.694	633	82.3

TABLE II. The diagonal approximation to the SoftAbs metric dramatically out-performs the full SoftAbs metric when applied to the funnel distribution, because the Hessian of the funnel potential is almost diagonal. As before, the step-size of the diagonal SoftAbs chain was determined with dual-averaging but with a target acceptance rate of $r = 0.8$.

RMHMC with Diagonal SoftAbs Metric

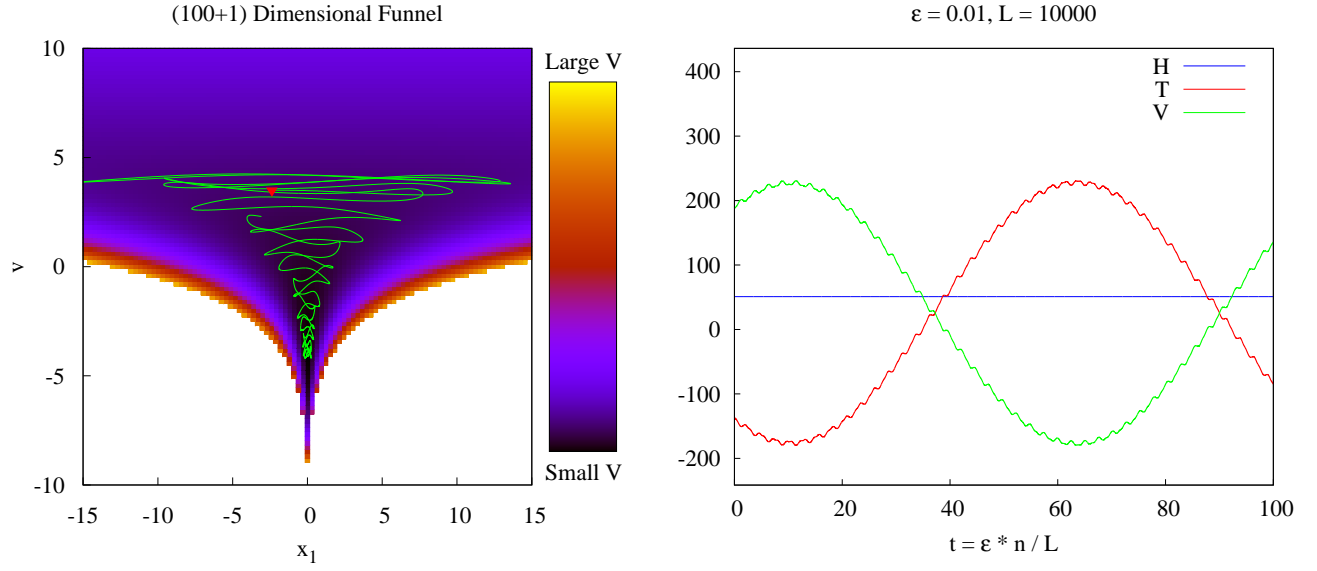


FIG. 7. Although clearly more ragged than the trajectories with the full SoftAbs metric, the trajectories with the diagonal SoftAbs metric explore almost the same expanse as the exact metric.

RMHMC with Diagonal SoftAbs Metric

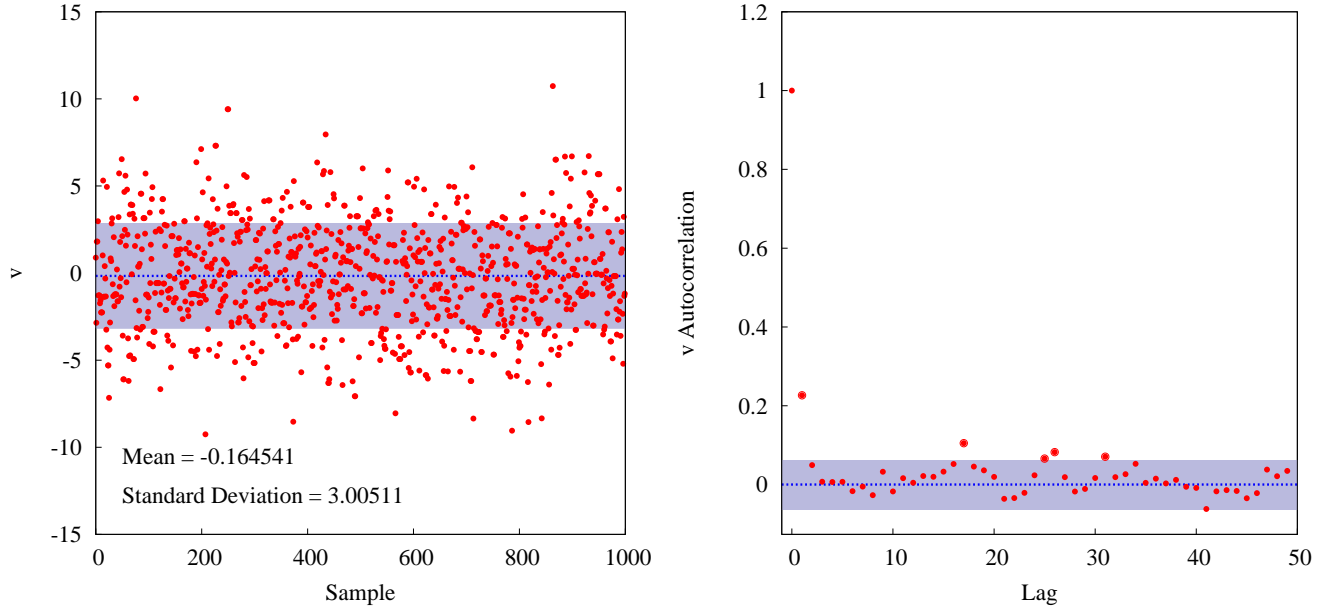


FIG. 8. The samples drawn with the diagonal SoftAbs metric are consistent with the true marginal for v , albeit at the expense of slightly higher autocorrelations.

RMHMC with Diagonal Outer-SoftAbs Metric

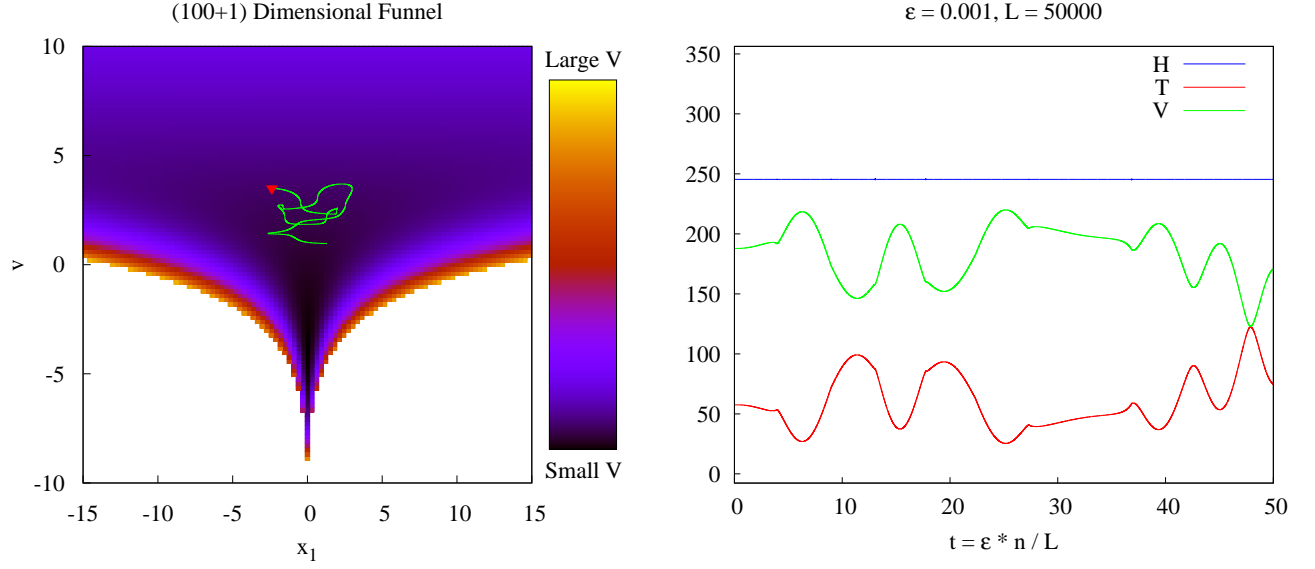


FIG. 9. When α has to be kept small to ensure the stability of the numerical evolution, the benefit of the dynamic metric is substantially reduced. The oscillating trajectories become much more complicated and, as in the EMHMC case, the variation of the potential over each trajectory is limited.

RMHMC with Diagonal Outer-SoftAbs Metric

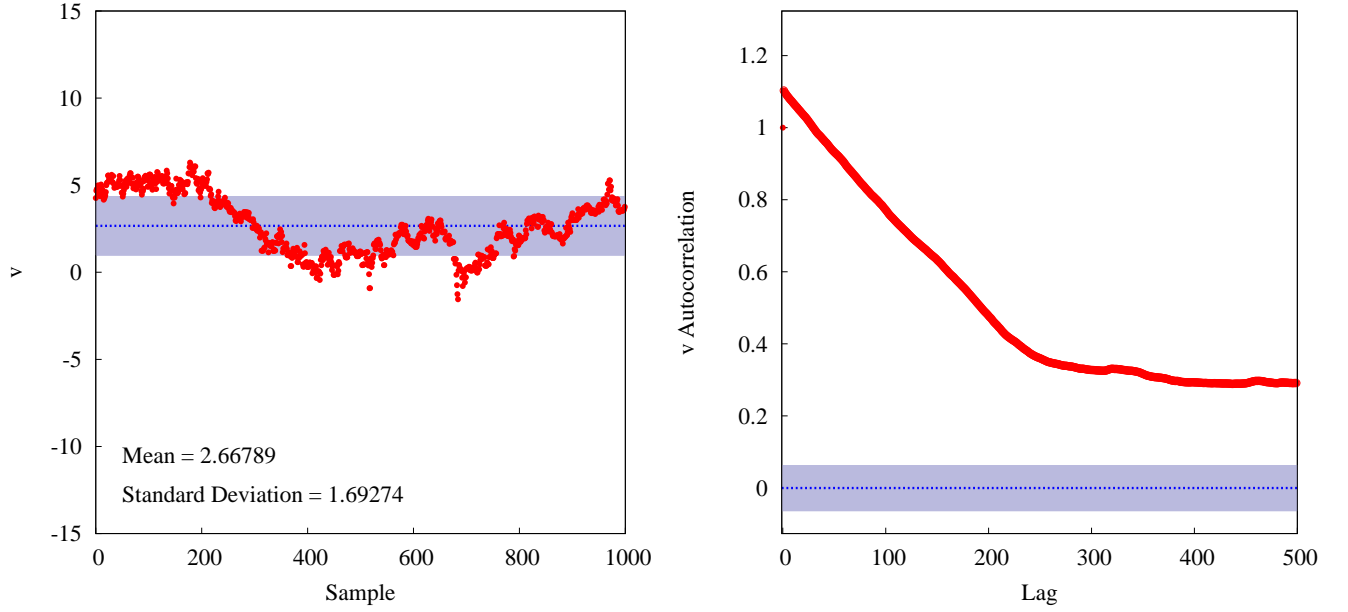


FIG. 10. The limited variation in potential induces random walk behavior in the diagonal outer-product approximation, while the small step-sizes required for stable evolution dramatically reduce the efficiency of each transition. Moreover, dual-averaging adaptation of the step-size introduces the same bias seen in adaptive tuning of EMHMC.

2. Outer-Product

A common approximation to the Hessian is the outer-product of the gradient [19],

$$\mathbf{H} \approx \mathbf{g}\mathbf{g}^T,$$

where $\mathbf{g} = \partial V$. Propagating this approximation through the SoftAbs map gives

$$\mathfrak{H} \approx \frac{(\mathbf{g} \cdot \mathbf{g})}{\sinh(\alpha \mathbf{g} \cdot \mathbf{g})} \left(\mathbb{I} + \frac{\cosh(\alpha \mathbf{g} \cdot \mathbf{g}) - 1}{(\mathbf{g} \cdot \mathbf{g})} \mathbf{g}\mathbf{g}^T \right).$$

Note that this is equivalent to the “background-score” metric proposed in [1] with functions scaling the rank-one update as well as the entire metric.

Evolution with this metric requires no derivatives beyond the Hessian, and with a careful implementation the computation scales as $O(N^2)^3$. The global coefficient, however, renders the metric almost singular in all but the simplest problems.

Along typical trajectories the norm of the gradient is far from zero, and the sinh becomes highly nonlinear. Consequently, the numerical evolution becomes unstable unless $\alpha \ll 1$ or $\epsilon \ll 1$, at which point the evolution becomes impractical. For example, with $\alpha = 1$ the step-size must be smaller than floating-point precision before the Metropolis accept rate rises above 0.5.

3. Diagonal Outer-Product

The severe non-linearities of the outer-product approximation can be avoided by considering only the diagonal elements of the outer-product,

$$\mathbf{H} \approx \text{Diag}(g_i^2),$$

which gives the metric

$$\mathfrak{H} \approx \text{Diag}(g_i^2 \coth \alpha g_i^2).$$

As above, $g_i = \partial_i V$.

Although this approximation avoids the highly-nonlinear coefficient above, the typically large values of the g_i^2 do induce large gradients in the Hamiltonian evolution. Taking $\alpha = 1$ yields stable trajectories, but the strong regularization reduces the log determinant’s ability to increase the variation in the potential (Figure 9) and limits the adaptability of the metric. The limited adaptability makes the Markov chain vulnerable to bias when adapting the integrator step-size and, consequently the samples exhibit the same random walk behavior and bias towards large v seen in the EMHMC case (Figure 10).

APPENDIX B: A SYMPLECTIC INTEGRATOR FOR RMHMC

One of the challenges of RMHMC is that the non-separable Hamiltonian requires a more sophisticated, and costly, symplectic integrator than EMHMC [20, 21]. In order to construct such an integrator first rewrite the Hamiltonian as

$$H = \tau + \phi,$$

where

$$\tau = \frac{1}{2} \mathbf{p}^T \cdot \Sigma^{-1}(\mathbf{q}) \cdot \mathbf{p}$$

and

$$\phi = \frac{1}{2} \log |\Sigma(\mathbf{q})| + V.$$

Both proper scalar functions, τ and ϕ motivate a natural splitting of the Hamiltonian which yields the integrator

$$\Phi_t = \hat{\phi}_{\frac{t}{2}} \circ \hat{\tau}_{\frac{t}{2}} \circ \hat{T}_t \circ \hat{\tau}_{\frac{t}{2}} \circ \hat{\phi}_{\frac{t}{2}}, \quad (2)$$

with

$$\begin{aligned} \hat{\phi} &= \frac{\partial \phi}{\partial q^i} \frac{\partial}{\partial p_i} \\ \hat{\tau} &= \frac{\partial \tau}{\partial q^i} \frac{\partial}{\partial p_i} \\ \hat{T} &= \frac{\partial \tau}{\partial p^i} \frac{\partial}{\partial q_i}. \end{aligned}$$

Of the individual operators, only $\hat{\phi}$ is separable and analytically calculable. The non-separable operators, $\hat{\tau}_{\frac{t}{2}} \circ \hat{T}_t \circ \hat{\tau}_{\frac{t}{2}}$, require the implicit generalized leapfrog algorithm (Algorithm 1). Efficient gradient computations complete the implementation (Algorithm 2).

³ In fact, with automatic differentiation techniques that can compute Hessian-vector products in linear time [19], the computation can be reduced further to $O(N)$.

$\mathbf{p} \leftarrow \mathbf{p} - \frac{\epsilon}{2} \partial \phi / \partial \mathbf{q}(\mathbf{p}, \mathbf{q})$ $\hat{\phi}$
 $\rho \leftarrow \mathbf{p}$ $\hat{\tau}_{\frac{1}{2}} \circ \hat{T}_t \circ \hat{\tau}_{\frac{1}{2}}$
while $\Delta p > \delta$ **do**
 $\mathbf{p}' \leftarrow \rho - \frac{\epsilon}{2} \cdot \partial \tau / \partial \mathbf{q}(\mathbf{p}, \mathbf{q})$
 $\Delta p = \max_i \{|p_i - p'_i|\}$
 $\mathbf{p} \leftarrow \mathbf{p}'$
end while

 $\sigma \leftarrow \mathbf{q}$
while $\Delta q > \delta$ **do**
 $\mathbf{q}' \leftarrow \sigma + \frac{\epsilon}{2} \cdot \partial \tau / \partial \mathbf{p}(\mathbf{p}, \sigma) + \frac{\epsilon}{2} \cdot \partial \tau / \partial \mathbf{p}(\mathbf{p}, \mathbf{q})$
 $\Delta q = \max_i \{|q_i - q'_i|\}$
 $\mathbf{q} \leftarrow \mathbf{q}'$
end while

 $\mathbf{p} \leftarrow \mathbf{p} - \frac{\epsilon}{2} \cdot \partial \tau / \partial \mathbf{q}(\mathbf{p}, \mathbf{q})$

 $\mathbf{p} \leftarrow \mathbf{p} - \frac{\epsilon}{2} \partial \phi / \partial \mathbf{q}(\mathbf{p}, \mathbf{q})$ $\hat{\phi}$

ALGORITHM 1. The spitting (2) admits a discrete integration of Hamilton's equations, approximating the evolution of the current position, \mathbf{q} , and momentum, \mathbf{p} , for a time, $\Delta t = \epsilon$. Because they are implicit, the first two steps of the generalized leapfrog must be solved with fixed-point iterations; in order to avoid position-dependent divergences causing bias in the Markov transitions, the iterations are continued until a given threshold instead of a set number of iterations.

Require: α *Regularization parameter*
Require: \mathbf{H} $\partial^2 V / \partial q^i \partial q^j$
Require: $\partial \mathbf{H}$ $\partial^3 V / \partial q^n \partial q^i \partial q^j$
Require: λ, \mathbf{Q} *Eigendecomposition of \mathbf{H}*

function $\partial \tau / \partial \mathbf{p}(\mathbf{p}, \mathbf{q})$
 return $\mathbf{Q} \cdot \lambda \lambda^{-1} \cdot \mathbf{Q}^T \cdot \mathbf{p}$

function $\partial \tau / \partial \mathbf{q}(\mathbf{p}, \mathbf{q})$
 $J_{ij} \leftarrow \frac{\lambda_i \coth \alpha \lambda_i - \lambda_j \coth \alpha \lambda_j}{\lambda_i - \lambda_j} (1 - \delta_{ij})$
 $\quad + \frac{\partial}{\partial \lambda_i} \lambda_i \coth \alpha \lambda_i \delta_{ij}$
 $\mathbf{D} \leftarrow \text{Diag}((\mathbf{Q}^T \cdot \mathbf{p})_i)$
 $\mathbf{M} \leftarrow \mathbf{Q} \cdot \mathbf{D} \cdot \mathbf{J} \cdot \mathbf{D} \cdot \mathbf{Q}^T$
 for $n = 1$ to N **do**
 $\delta_n \leftarrow \frac{1}{2} \text{Tr}[-\mathbf{M} \cdot \partial_n \mathbf{H}]$
 end for
 return δ

function $\partial \phi / \partial \mathbf{q}(\mathbf{p}, \mathbf{q})$
 $J_{ij} \leftarrow \frac{\lambda_i \coth \alpha \lambda_i - \lambda_j \coth \alpha \lambda_j}{\lambda_i - \lambda_j} (1 - \delta_{ij})$
 $\quad + \frac{\partial}{\partial \lambda_i} \lambda_i \coth \alpha \lambda_i \delta_{ij}$
 $\mathbf{R} \leftarrow \text{Diag}\left(\frac{1}{\lambda_i \coth \alpha \lambda_i}\right)$
 $\mathbf{M} \leftarrow \mathbf{Q}(\mathbf{R} \circ \mathbf{J})\mathbf{Q}^T$
 for $n = 1$ to N **do**
 $\delta_n \leftarrow \frac{1}{2} \text{Tr}[\mathbf{M} \cdot \partial_n \mathbf{H}] + \partial_n V$
 end for
 return δ

ALGORITHM 2. With careful manipulation and caching of intermediate matrices, the gradients necessary for Hamiltonian evolution (Algorithm 1) can be computed in $O(N^3)$. Note that each trace can be computed in $O(N^2)$ as $\text{Tr}[\mathbf{A} \cdot \mathbf{B}] = \sum_{i,j} A_{ij} B_{ji}$.

-
- [1] M. Betancourt and L. C. Stein, (2011), arXiv:1112.4118 [stat].
 - [2] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth, *Physics Letters B*, **195**, 216 (1987), ISSN 0370-2693.
 - [3] R. Neal, in *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng (CRC Press, New York, 2011).
 - [4] M. Girolami and B. Calderhead, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **73**, 123 (2011), ISSN 1467-9868.
 - [5] S. Amari and H. Nagaoka, *Methods of information geometry*, Vol. 191 (Amer Mathematical Society, 2007).
 - [6] M. Spivak, *A Comprehensive Introduction to Differential Geometry*, Vol. 1 (Publish or Perish, Inc, Houston, Texas, 2005).
 - [7] M. Celis, J. E. Dennis, and T. R. A., in *Numerical Optimization 1984*, edited by P. Boggs, R. Byrd, and R. Schnabel (SIAM, Philadelphia, 1985).
 - [8] C. Moler and C. Van Loan, *SIAM review*, **45**, 3 (2003).
 - [9] K. Aizu, *Journal of Mathematical Physics*, **4**, 762 (1963).
 - [10] R. M. Wilcox, *Journal of Mathematical Physics*, **8**, 962 (1967).
 - [11] J. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics* (Wiley, New York, 2007).
 - [12] R. Neal, *Annals of statistics*, 705 (2003).
 - [13] A. Gelman, J. Carlin, H. Stern, and D. Rubin, *Bayesian Data Analysis* (Chapman & Hall/CRC Press, Boca Raton, Florida, 2004).
 - [14] I. Murray and R. Prescott Adams, ArXiv e-prints (2010), arXiv:1006.0868 [stat.CO].
 - [15] M. D. Hoffman and A. Gelman, ArXiv e-prints (2011), arXiv:1111.4246 [stat.CO].
 - [16] C. Geyer, in *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng (CRC Press, New York, 2011).
 - [17] M. Spivak, *A Comprehensive Introduction to Differential Geometry*, Vol. 2 (Publish or Perish, Inc, Houston, Texas, 2005).
 - [18] B. Chow and D. Knopf, in *Mathematical Surveys and Monographs 110* (American Mathematical Society, Providence, 2004).
 - [19] C. Bishop, *Pattern Classification and Machine Learning* (Springer, New York, 2007).
 - [20] B. Leimkuhler and S. Reich, *Simulating Hamiltonian Dynamics* (Cambridge University Press, New York, 2004).
 - [21] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration* (Springer, New York, 2006).